

- يرجى الاعتماد على الـ Slides في مذاكرة الجزء النظري في البداية
- السلايد الخضراء من 1 إلى 21
- سلايد المرجع من 40 إلى 57

Feature	8051	8052	8031
ROM	4K	8K	0K
RAM	128	256	128
Timers	2	3	2
I/O pins	32	32	32
Serial port	1	1	1
Interrupt source	6	8	6

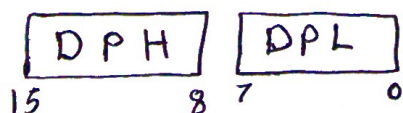
8051 Registers

1 8-bit registers

- * A (Accumulator)
- * B, R0, R1, R2, R3, R4, R5, R6, R7

2 16-bit registers

- * PC (Program Counter)
 - point to the next instruction to be executed
- * DPTR (Data pointer)
 - point to Location in ROM or External RAM
 - Could be divided to



3 Special Function Registers

- * A, B, DPTR, PC
- * P0, P1, P2, P3
- * SP (stack pointer)
- * etc.

MOV Instruction

- It's used to copy data from Source to the destination
- MOV destination, Source
- The Source Can be intermediate data by using ~~X~~

MOV A, ~~X~~ 55H → Hex

MOV A, ~~X~~ 15 → Decimal

MOV A, ~~X~~ 'B' → ASCII

MOV A, ~~X~~ 0000 1111B → Binary

- The used data must be suitable in size to destination

MOV A, ~~X~~ 355H invalid X

MOV A, ~~X~~ 300 invalid X

MOV A, ~~X~~ 33H Valid ✓

MOV DPTR, ~~X~~ 35FFH valid ✓

- You should put 0 in the start of Hex-number if it start with letter

MOV A, ~~X~~ F0H invalid X

MOV A, ~~X~~ 0F0H valid ✓

- You could not move data from Registers R0:R7 between each other

MOV R0, R2 invalid X

MOV A, R0 valid ✓

ADD Instruction

ADD A, R0 → A = A + R0

- The destination must be the Accumulator A

Assembly Statements

- 1 Assembly Language Instructions
 - Tell CPU what to do
- 2 Directives
 - Give directions to assembler

Assembly Instructions Format

[label:] Mnemonic [operands] [; comment]

- Steps to Create Assembly Program slides 67 → 70

* 1st (List) File

- Lists all opcodes, addresses and errors
- It's used to find syntax errors and debugging

* hex File

- It's the program ready to be burnt into ROM

* When 8051 is powered-up the program counter (PC) has the value 0000H

* The ROM in 8051 is 4KB which can be accessed using address between 0000H → 0FFF

Assembly Directives

- 1 ORG
 - indicate the start of address
- 2 END
 - indicate the end of the assembly program

3 EQU

- Used to define constant without reserving location in the memory

4 DB

- The most common data directive used in 8051
- Used to reserve and store data in ROM

5 BIT

- Used to point to special function bits
- The EQU can do the same job

Example

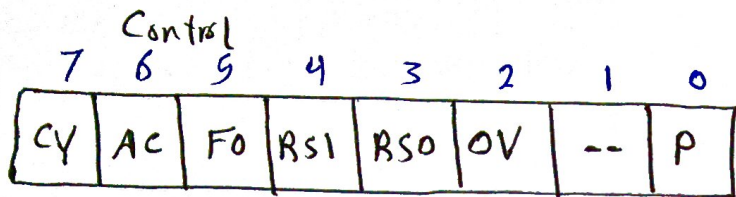
```
COUNT EQU *50
BIT
LEDA BIT P0.1
LEDB EQU P0.2
no colon ORG 0 → Code start at 0
MOV A, *COUNT
MOV C, LEDA
MOV A, DATA
ORG 500H → Start at 500H
DATA: DB 8
DATA: DB 50H
END
```

0500	08
0501	50
Address	Data

ROM

Program Status Word

- PSW is a 8-bit register used in conditions and flow



- CY → Carry from D7
- AC → Auxiliary Carry from D3 to D4
- FO → User defined
- RS1 → Register Bank Selectors
- RS0 → Register Bank Selectors
- OV → Over Flow
- → User defined
- P → Parity → if 1's is odd → P=1
→ if 1's is even → P=0

- Look At Examples in slides
87 → 89

8051 RAM

- It's 128-Byte RAM
- Address from 00 to 7F

[1] Register Bank 0 [00 → 07]

- Default Location of Register R0, R1, ..., R7

[2] Register Bank 1 [08 → 0F]

- Default Location for Stack
- SP → points to the last used location for stack

Default SP = 07H

[3] Register Bank 2 [10 → 17]

[4] Register Bank 3 [18 → 1F]

[5] Bit Addressable RAM [20 → 2F]

[6] Scratch Pad RAM [30 → 7F]

Note:

The Addresses from 20 to FF is reserved for Special Function Registers

Some Bit Instructions

SETB	C	→ C = 1
CLR	PSW.4	→ RS1 = 0
CPL	PI.0	→ Complement → NOT

Stack Operations

[1] PUSH 5

- increment SP
- Copy data from memory location 5 (R5) to the memory location with Address stored in SP

[2] POP 5

- Copy data from top of Stack to location 5 (R5)
- Decrement SP

- Look At Examples in slides
98 → 103